

# **Artificial Neural Networks with Application to the Sacramento-San Joaquin Delta**

California Department of Water Resources  
Delta Modeling Section, Division of Planning

Ralph Finch<sup>1</sup> and Nicky Sandhu<sup>2</sup>  
March 8, 1995

## **Background**

The need for a fast, reasonably accurate method of modeling the relationship between flows and water quality in the Sacramento-San-Joaquin Delta has been recognized for many years. Uses are to estimate the flows required to achieve a certain salinity standard (the so-called MDO routine, used in DWR's statewide planning simulation model DWRSIM), to rapidly simulate many different flow/salinity scenarios for optimization, real-time control of project operation, and so on.

Previous efforts have used a simple conceptually-based representation of the Delta (MDO), or linear or simple nonlinear equations. Recently, the Department of Water Resources, Delta Modeling Section, has investigated a new mathematical technology called Artificial Neural Networks (ANNs) as an alternative to conventional techniques. ANNs may be thought of as a kind of multiple nonlinear regression. ANNs offer several advantages over previous methods: they are nonlinear (and do not require that the transfer function between inputs and outputs be predetermined), they are setup with only very broad assumptions about Delta hydrodynamics, and they easily accept multiple inputs of different kinds (e.g. flows and gate positions). These advantages are discussed in more detail below. Their only significant disadvantage is that they tend to require more data to calibrate than conventional methods. However in the case of the Delta, there is sufficient data (about 25 years of daily data) to calibrate and validate ANNs for a variety of purposes.

To illustrate the power and flexibility of ANNs, we have developed a set of calibration and validation sets on 20 years of historical data from the Delta. The data used in all the figures are daily values, smoothed with a 14-day moving average, and subsampled every 7 days. We calibrated on the period 1981-1991, and validated each model with data from 1971-1981. The calibration period was chosen so as to cover a wide range of salinity values and Delta flow conditions. Later we will show the importance of never performing only a calibration over all available data, but rather splitting the data into calibration and validation sets.

---

<sup>1</sup>Senior Engineer, W.R. / 916-653-8268 / rfinch@dop.water.ca.gov

<sup>2</sup>Engineer, W.R. / 916-653-7552 / nsandhu@dop.water.ca.gov

The ANNs used are from the public domain program Stuttgart Neural Network Simulator, which runs under Unix, available on the Internet. Other public domain or shareware codes are available for PCs, and MATLAB offers a Neural Network Toolbox which is helpful for learning and experimenting with a variety of different ANN architectures.

We first illustrate the well-known deficiencies of linear and quasi-nonlinear models with Figure 1. A simple multiple linear regression was fitted to the log-transformed calibration data (Collinsville EC and Net Delta Outflow), and the resulting model applied to the validation data. This model (and all models presented here) use flow memory, that is, the input is composed of current flow and previous flows, in this case back to 30 weeks. While a better model could be fitted on just the low flow, high salinity data, it is apparent from both the time-series and scatter plots that linear models are insufficient to model the Delta. By comparison, Figure 2, an ANN developed with NDO flow input, Collinsville EC salinity output, and memory applied to the inputs, fits much better than the simple model shown in Figure 1.

A significant advantage of ANNs with respect to the Delta is that only very broad, general assumptions need be made about Delta dynamics. For instance, we simply assume that there is some kind of relationship in the Delta between flows and gate positions, and salinities. ANNs do not require prejudging the importance of various flows, the significance of gates or channels, or the existence of such things as carriage water. Rather, ANNs can handle a variety of inputs, and will automatically determine which are relevant and which are not. For instance, Net Delta Outflow (NDO) is a calculated number comprised of inflows (Sacramento and San Joaquin Rivers, Eastside Streams, Yolo Bypass, and Delta precipitation) minus the outflows (Delta consumptive use, exports, and diversions). It is obvious that under two different Delta flow hydrologies the same number for NDO could be calculated but with quite different inflows and outflows. These different hydrologies could result in very different salinity patterns in the Delta, since salts are transported by water movement, and water from different sources has different salt concentrations.

To illustrate this, we developed an ANN model, again to estimate Collinsville EC, but this time instead of NDO we used the individual flows that comprise NDO and Cross Channel Gate position information. The result is given in Figure 3. Some improvement is noted, in both the time-series and the scatter plots, and also in the Mean-Square-Error (MSE) number shown in the upper left corner of the time-series plot. We reduced the memory in Figure 3 from 30 weeks to 10 weeks to save calibration time; more improvement might be noted had the memory been kept at 30 weeks.

The importance of using multiple inputs, instead of NDO only, is even clearer with a model to estimate chlorides in the interior Delta, at Contra Costa Canal Pumping Plant #1. Figure 4 is the result of an ANN applied to that location, using NDO with memory as its only input. Results are poor. However, when multiple flow and gate positions are used as input, the results improve significantly, as shown in Figure 5, even though the memory is again reduced to save calibration time.

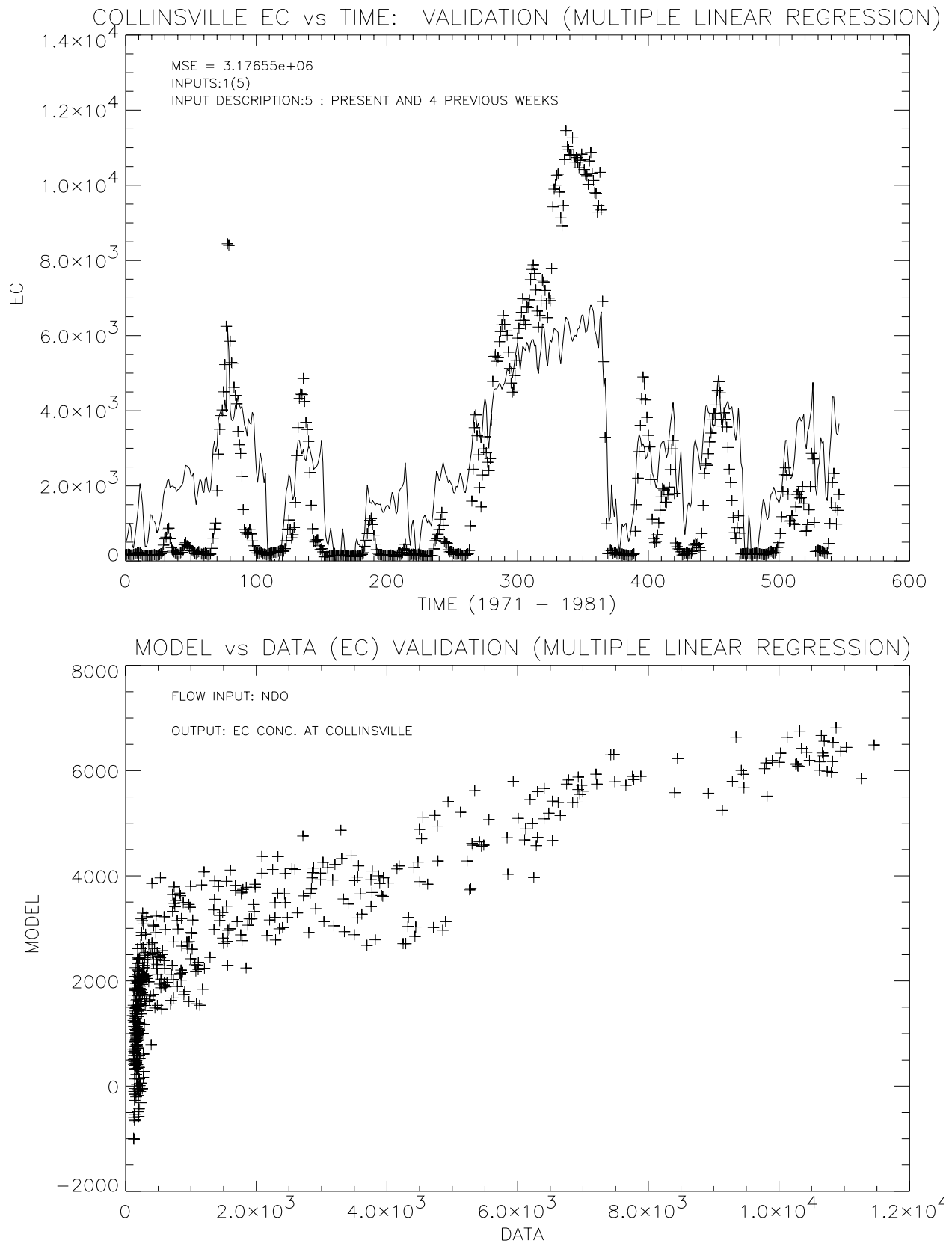
We finish the ANN examples by showing the effects of over-fitting during calibration. Figure 6 is the calibration plot of the ANN model of Figure 3 (Collinsville EC, component flows). Figure 7 is the calibration results using the same data and model, except we increased the number of neurons (internal fitting functions) from 6 to 22. The numbers of degrees of freedom increased from 542 to 1802, about 3 times the number of data points. The calibration plot (Figure 7) shows a marked improvement over the same ANN with less neurons (Figure 6). However, the validation plot, Figure 8, shows a decrease in accuracy compared to the validation of the properly fitted model (Figure 3). Thus, it is important when discussing the merits of numerical models to perform comparisons using validation data, and not just calibration.

## **Future Directions**

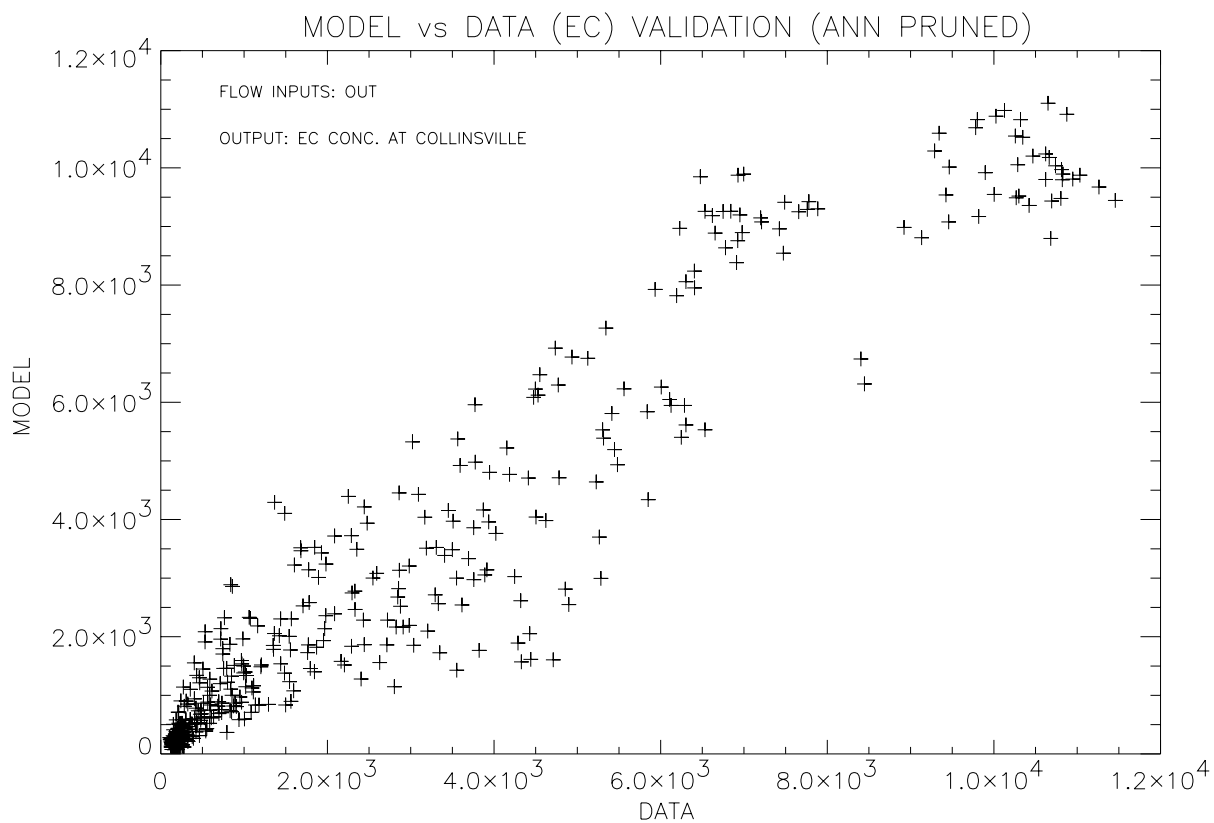
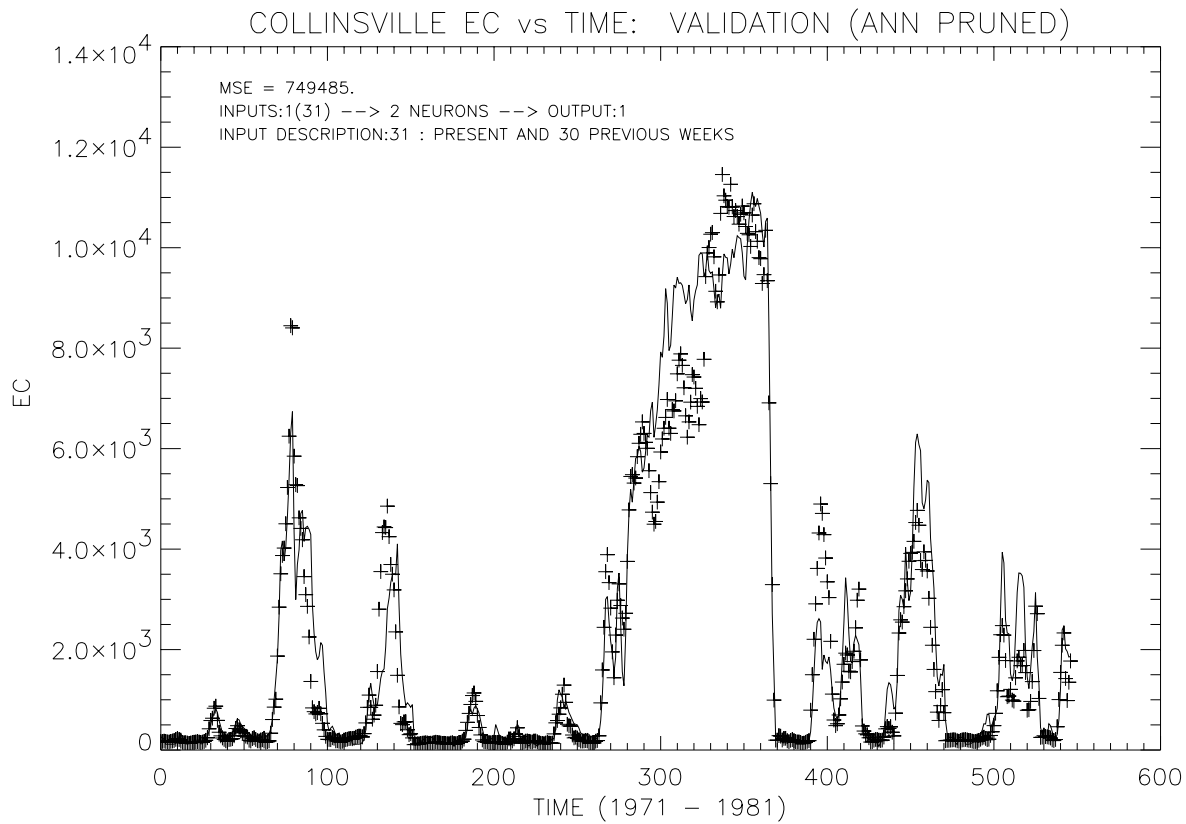
In the immediate future we plan on trying different ways of representing Delta flows, and performing tests to see which flow inputs are important under what conditions for estimating salinity. Also, we will develop dynamic error estimates of the model output salinity, since it is critical that flow requirements to meet salinities be stated on a probabilistic, not deterministic, basis.

For Delta planning simulations, for instance, where the effects of barriers placed in the Delta must be modeled, a numerical model such as the Delta Simulation Model could be run, and ANNs calibrated on the numerical results. ANNs could be used to replace the current MDO routine in DWRSIM, and by using an ANN trained on simulated salinities, would offer the advantage of a systematic way of running DWRSIM with proposed changes to the Delta. A reverse ANN, where Sacramento flows would be calculated given the required salinities and other flows, might be most appropriate.

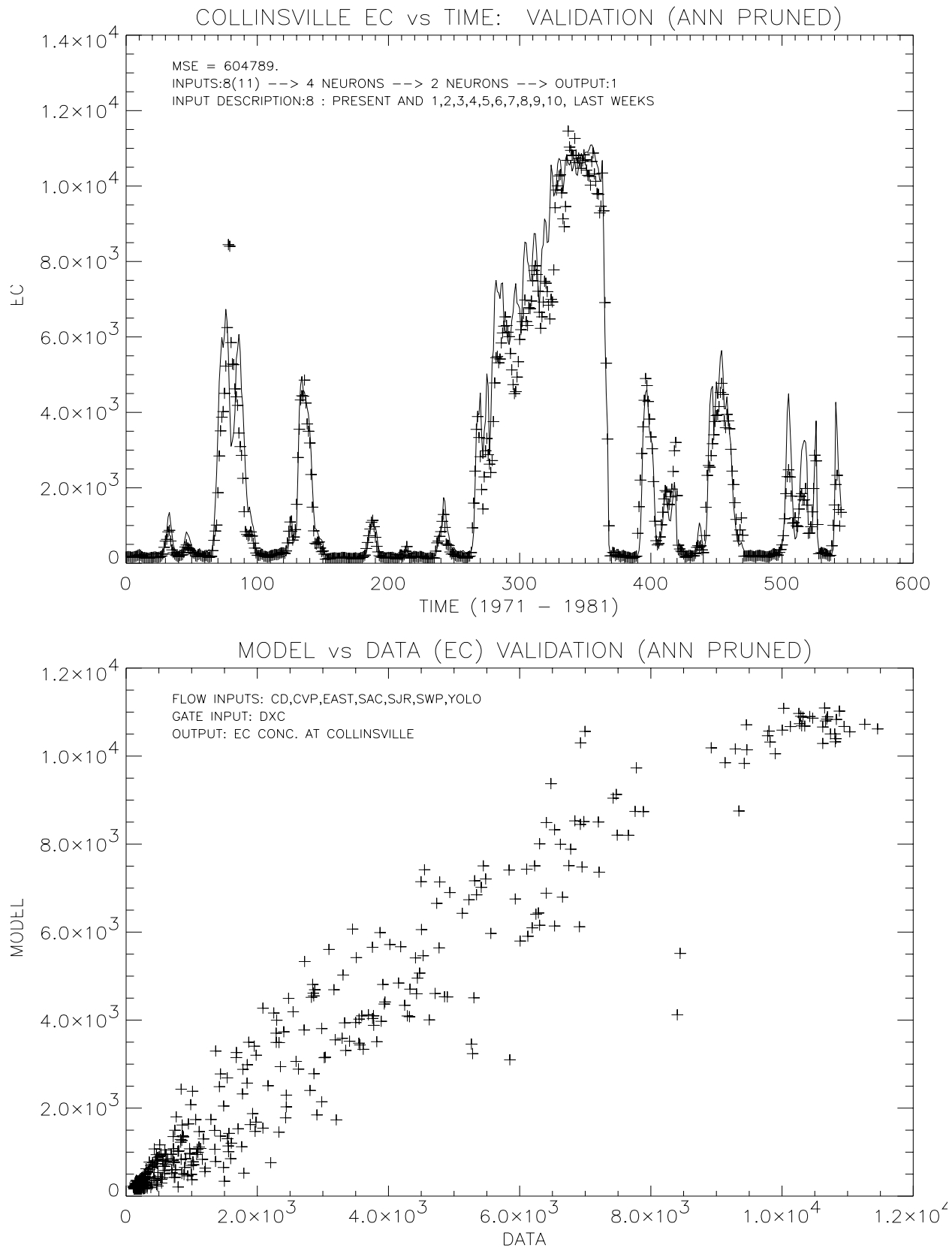
For the longer term we will try different ANN models such as recurrent and time-delay, and test ANNs in a variety of Delta tasks such as THM modeling, filling in missing data, converting between different water quality units such as EC, TDS, and ion concentrations, and adapting the model for real-time simulation and prediction.



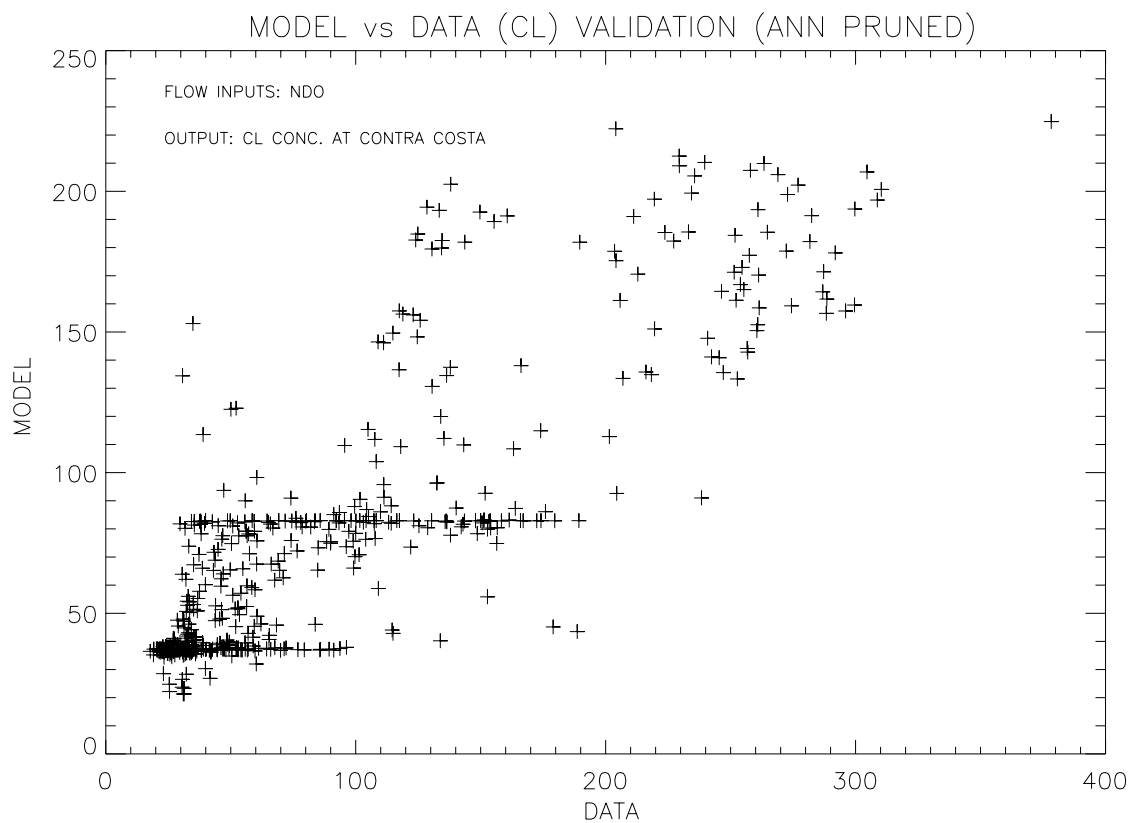
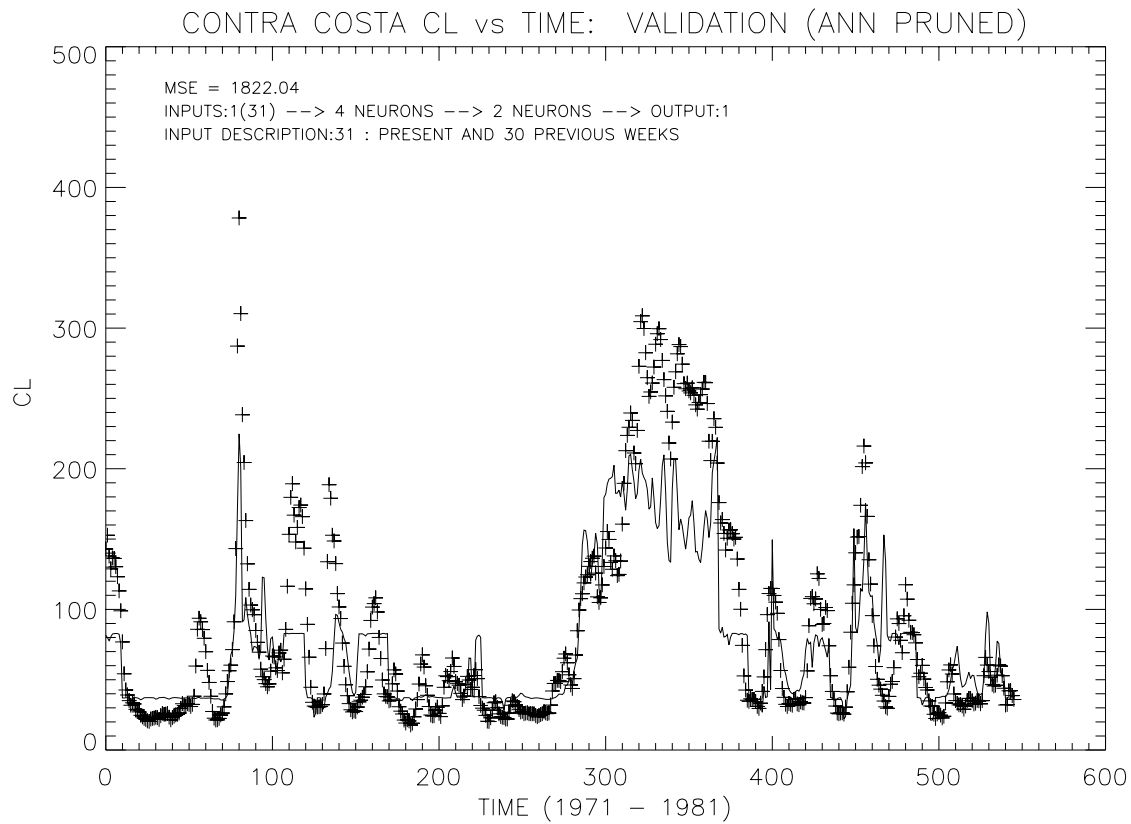
**Figure 1**



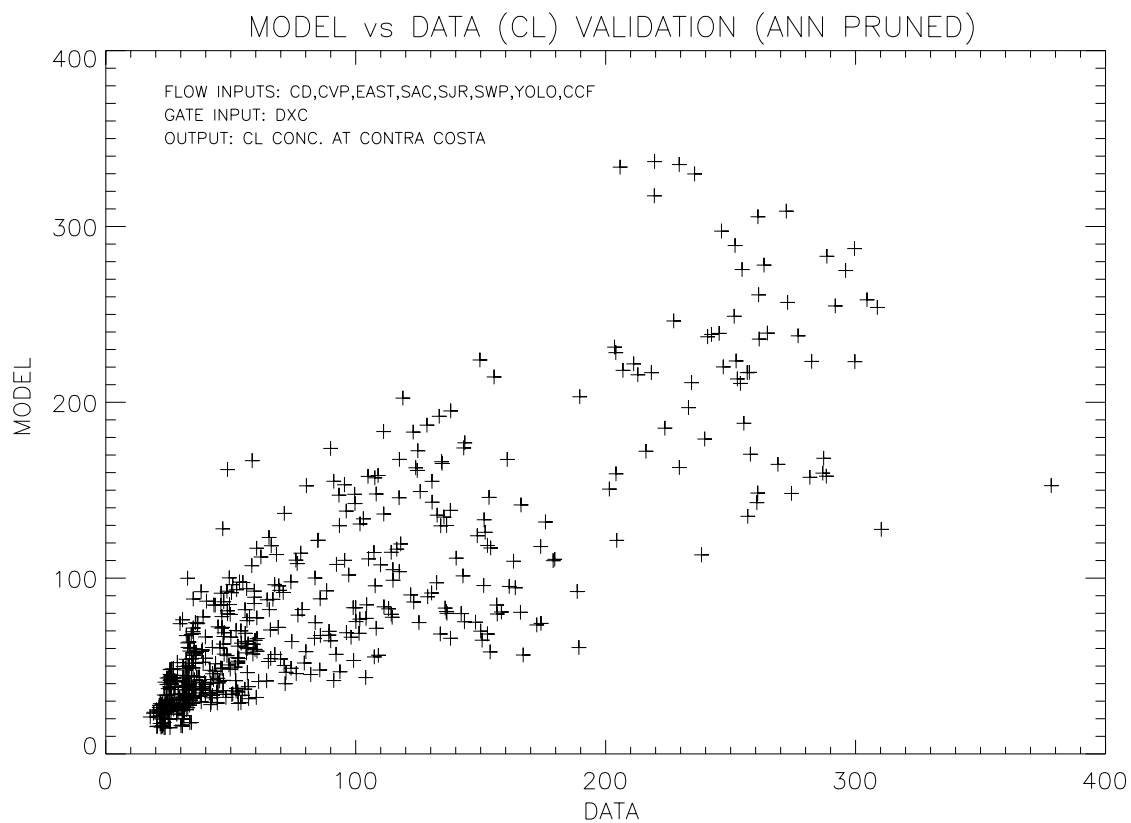
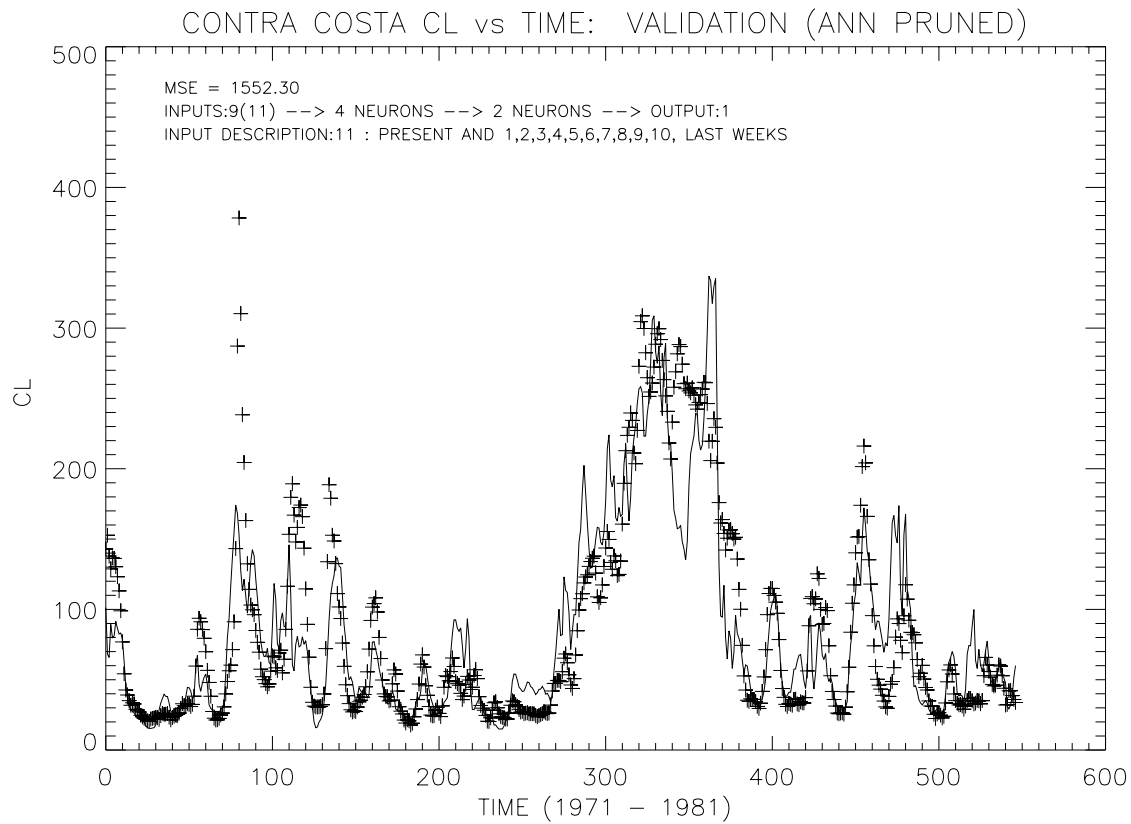
**Figure 2**



**Figure 3**

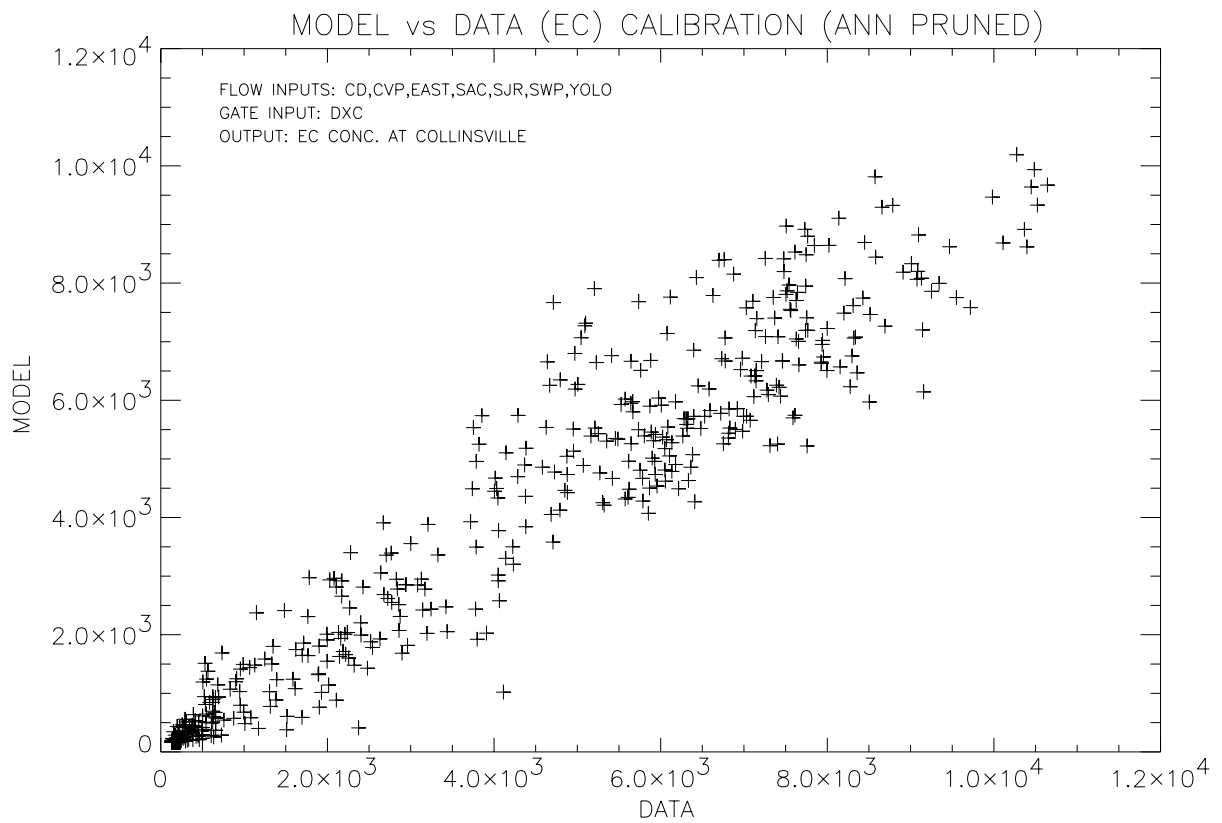
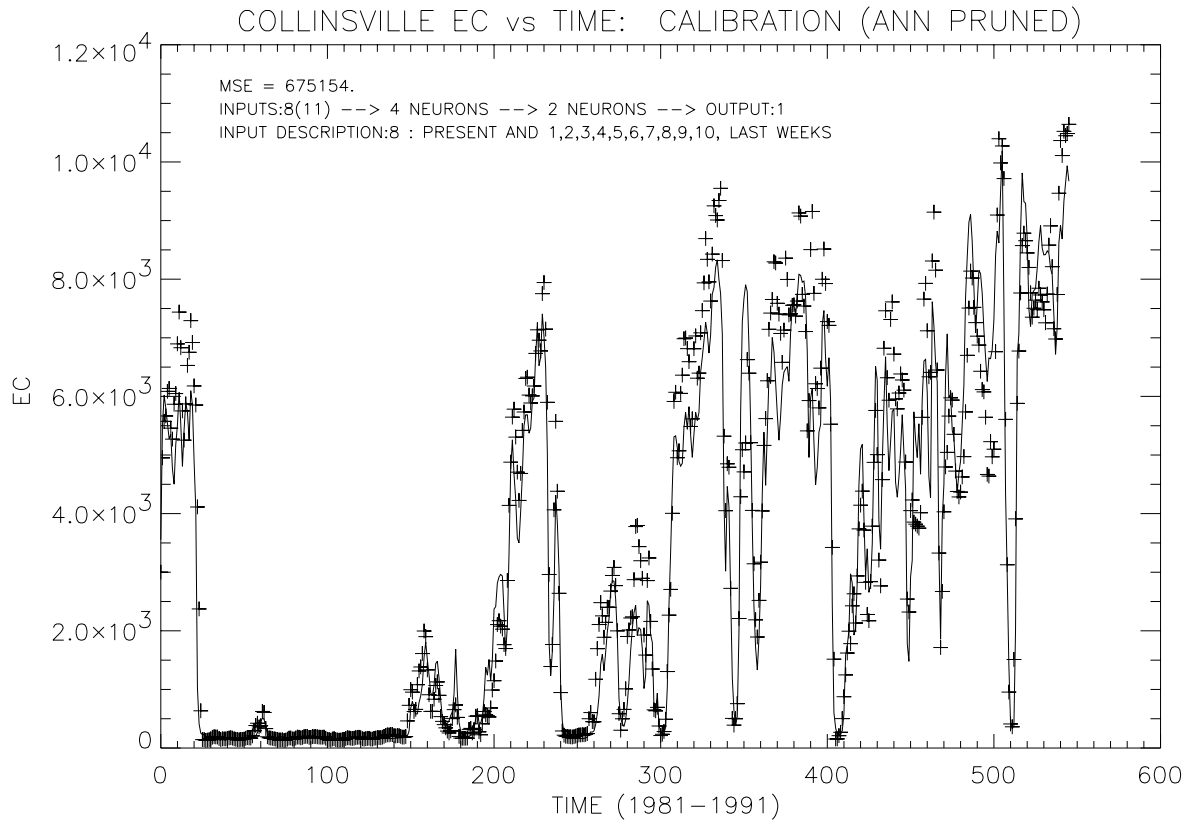


**Figure 4**

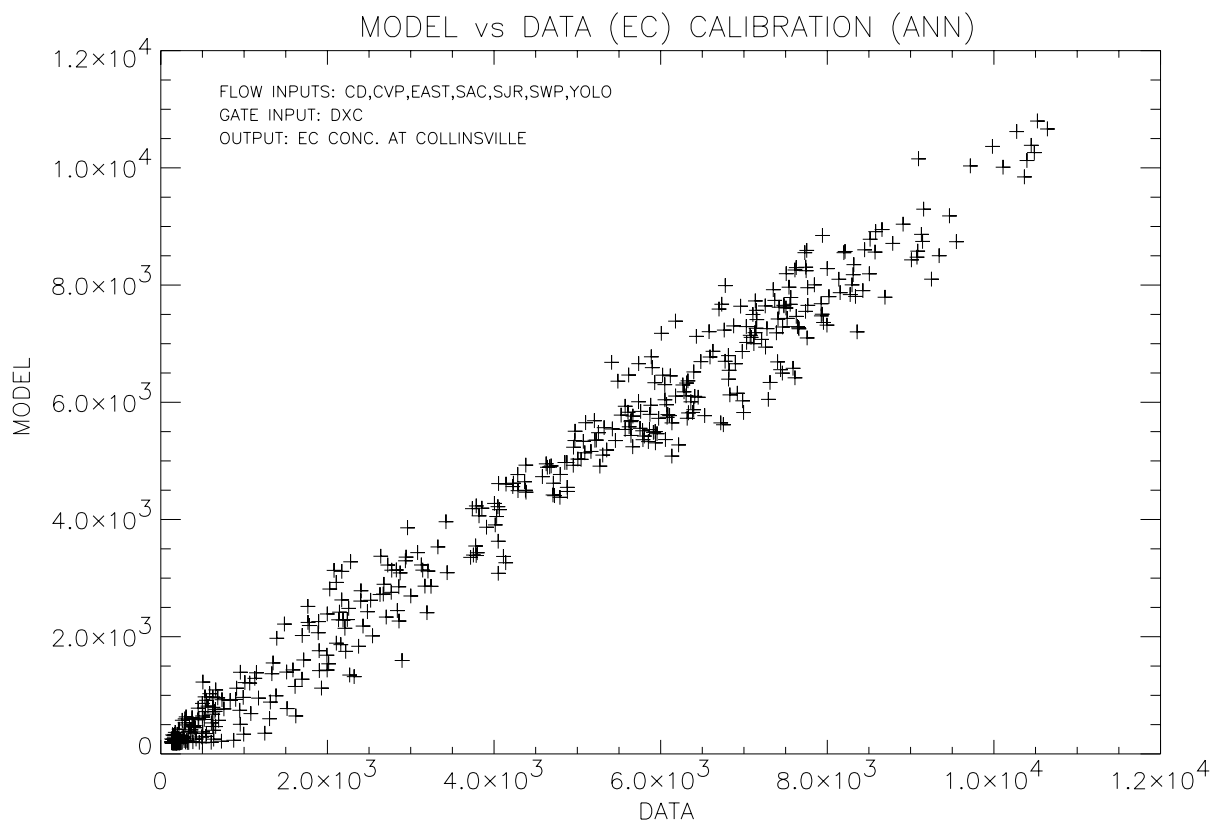
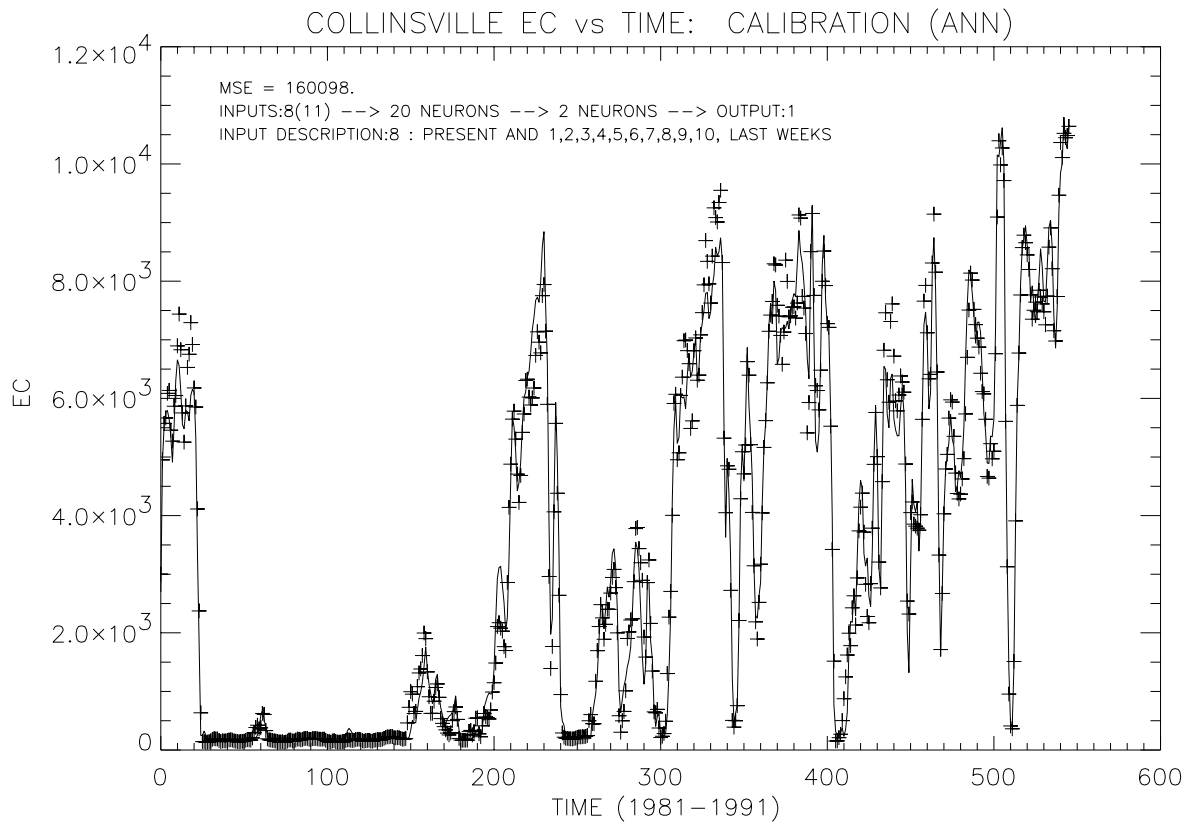


**Figure 5**

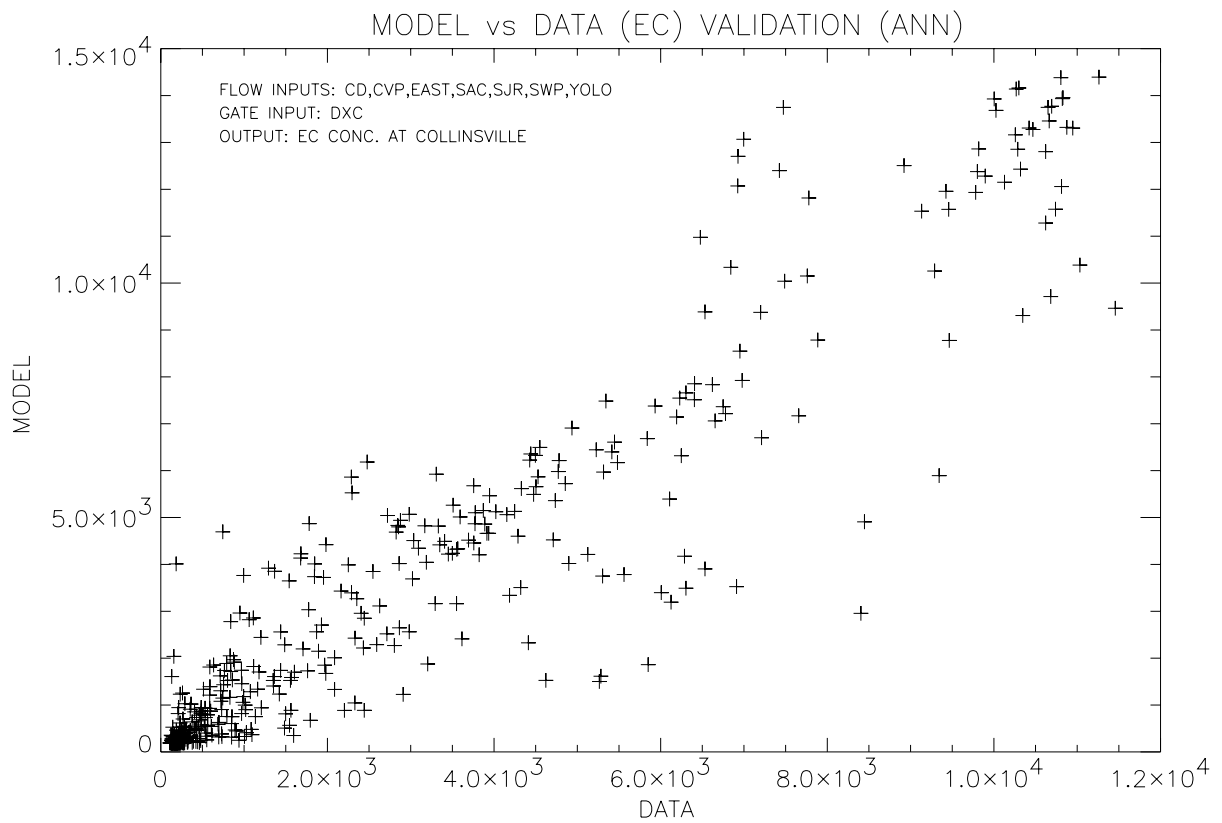
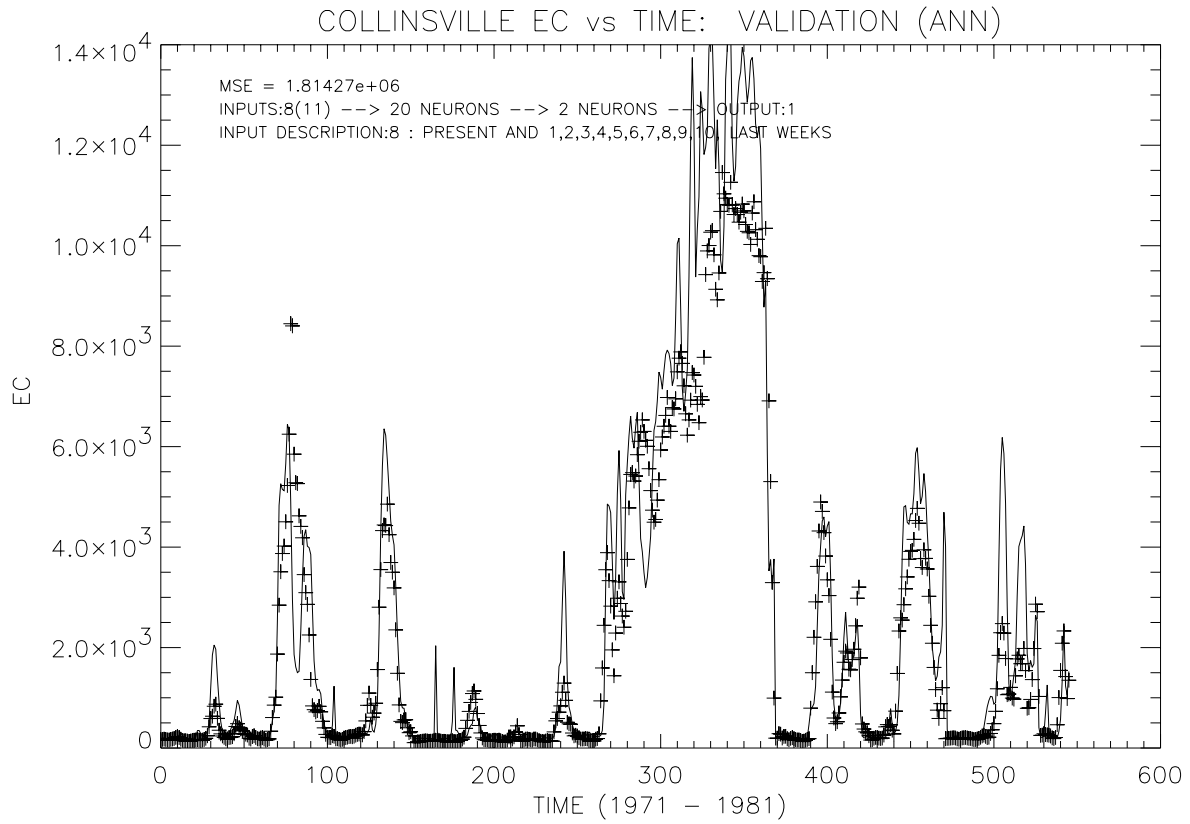




**Figure 6**



**Figure 7**



**Figure 8**